Two-Finger 3D Rotations for Novice Users: Surjective and Integral Interactions

Élisabeth Rousset, François Bérard, Michael Ortega

Two-Finger 3D Rotations for Novice Users: Surjective and Integral Interactions

Élisabeth Rousset, François Bérard, Michaël Ortega
Univ. Grenoble Alpes, LIG, F-38000 Grenoble, France
CNRS, LIG, F-38000 Grenoble, France
first.last@imag.fr

ABSTRACT

Now that 3D interaction is available on tablets and smart phones, it becomes critical to provide efficient 3D interaction techniques for novice users. This paper investigates interaction techniques for 3D rotation with two fingers of a single hand, on multitouch mobile devices.

We introduce two new rotation techniques that allow integral control of the 3 axes of rotation. These techniques also satisfy a new criterion that we introduce: surjection. We ran a study to compare the new techniques with two widely used rotation techniques from the literature. Results indicate that surjection and integration lead to a performance improvement of a group of participants who had no prior experience in 3D interaction. Qualitative results also indicate participants' preference for the new interaction techniques.

Categories and Subject Descriptors

H.5.m. [**Information Interfaces and Presentation**]: User Interfaces - Interaction styles, Input devices and strategies

General Terms

3D interaction, 3D rotation, multitouch

Keywords

3D interaction, 3D rotation, multitouch

1. INTRODUCTION

Interacting with 3D scenes (3D interaction in short) used to require dedicated hardware and was thus limited to expert users (i.e. professionals or enthusiasts) with explicit needs. Today, even mobile devices such as tablets and smart phones are equipped with powerful graphic processors. This makes the rendering of interactive 3D content almost as easy to implement as 2D content. From this, we can expect 3D content to become widespread and to be presented to novice users, i.e. users with no specific training with computers and 3D interaction. In parallel, multi-touch interaction is becoming the standard on mobile devices. The challenge is thus to provide intuitive multi-touch 3D interaction to the novice users.

Some web-stores, for example, may chose to replace 2D pictures of their items with their 3D counterparts. The benefit is to allow users to observe the items from any angle and distance that they want. Such a task is called an "observation task", and can be performed with only 3D rotations of the object. Observation tasks represent a small subset of all the tasks covered by "3D interaction", but this subset is fundamental. Indeed, 3D interaction in general is difficult and requires significant efforts from users [7], especially on mobile devices where users typically hold the device with one hand and can only interact with the fingers of the other hand. Attempting to offer full 3D interaction to the novice users on mobile devices seems impractical, and probably irrelevant. Efforts should focus, at least initially, on the design of intuitive and efficient interactions for *observation tasks*, and thus for 3D rotations.

Several interaction techniques for the control of 3D rotations have been presented in the literature. These techniques can be compared using well established criteria, such as transitivity [1] and integrality [7]. In this paper, we introduce the new criterion of *surjection*. An interaction technique is surjective if it offers the possibility to reach any final state of the controlled object from any current state of an ongoing interaction of this technique. We explain how surjection is a key criterion to allow a smooth transition between the different phases of an interaction.

In this paper, after reviewing the literature about interaction techniques for 3D rotations, we present the surjection criterion. We analyze the state of the art in the light of surjection and other criteria. This analysis leads us to the introduction of two new interaction techniques which satisfy an increased number of criteria compared to previous work. We then report on a user experiment aimed at comparing user performances using these two techniques. We discuss the result of the experiment and conclude by reflecting on the implications of this work.

AVI '14, May 27 - 30, 2014, Como, Italy
Copyright 2014 ACM 978-1-4503-2775-6/14/05…$15.00.
http://dx.doi.org/10.1145/2598153.2598183

2. RELATED WORK

Our study is closely related to 3 bodies of research: 3D rotations, multi-touch 3D interaction and the integral control of multiple degrees of freedom.

2.1 3D Rotations

Chen *et al.* studied different interaction techniques using a simple mouse [2]. They were the first to propose the emulation of a trackball as an alternative to sliders to perform 3D rotations. They demonstrated that the trackballs *Virtual Sphere* and *Continuous XY+Z* were faster than sliders for complex rotations, with no measurable loss of precision. Later, Shoemake proposed the *Arcball* technique, and presented it as more user-friendly than the Virtual Sphere [12]. According to Shoemake, the Arcball technique uses a better matching between human factors and mathematical fundamentals. The technique avoids the "hysteresis effect", i.e. it brings the object back to its initial orientation when the user closes a mouse motion loop. However, Hinckley *et al.* compared the efficiency of the Virtual Sphere and the Arcball, and found no significant difference on performance between them [4].

More recently, Bade *et al.* introduced ergonomic principles to compare rotation techniques, in addition to more traditional performance measurements [1]. They proposed the four principles:

1. $B1$: Similar actions should provoke similar reactions.
2. $B2$: Direction of rotation should match the direction of pointing device movement.
3. $B3$: 3D rotation should be transitive.
4. $B4$: The control-to-display (C/D) ratio should be customizable.

Analysing the existing techniques with these principles, Bade *et al.* showed that the Continuous XY+Z (renamed *Two Axis Valuator*) matches most of them, unlike the Virtual Sphere and the Arcball.

Thus far, the Two Axis Valuator has been presented as the most user-friendly virtual trackball. All these trackball-like techniques, however, have been developed for 2 DOF devices (e.g. a mouse). As a consequence, different rotation behaviors are triggered by spatial activation: (1) the pure Z rotation is a mode that is triggered when the cursor is out of the trackball, and (2) the Z rotation is not dissociated from the X and Y rotations when the cursor is into the sphere projection.

Zhao *et al.* proposed the use of the third DOF of the mouse (the wheel) to clearly dissociate XY rotations from Z rotations, and to dissociate the cursor position from mode selection [14]. The wheel is used for Z rotation only, while XY are controlled by the cursor position. With this approach however, there is a discrepancy between the physical rotation axis of the wheel, which is aligned with the X axis of the screen, and the logical rotation axis which is orthogonal to the screen.

2.2 Multi-touch 3D rotations

With the wide availability of direct multi-touch displays, research has been focused on new interaction techniques that leverage the use of multiple fingers in order to increase the number of input DOF. Hancock *et al.* studied multitouch for 3D interaction in general, and proposed *Sticky Tools*, a full 6 DOF interaction technique, thus including 3D rotations [3]. However, the technique was designed for large multi-touch displays: 3 DOF rotations are achieved with 3 fingers from two hands, which is not suitable for single-hand interaction on mobile devices. Reisman *et al.* proposed another 6 DOF multitouch manipulation technique, called *Screenspace*, which principles is to move the manipulated object so that the contact point of the fingers remain stuck to the object [9]. They found that Screenspace was more efficient than StickyTools, but it was not always intuitive and sometimes resulted in unpredictable behaviors, which makes it inappropriate for novice users. In addition, its principle dictates that some orientations of the manipulated object are not reachable in a single gesture: those where a control point should go to "the back" of the object.

Liu *et al.* proposed a 6 DOF manipulation technique that combines the well established two-finger 2D interaction for Rotation, Scaling and Translation (RST), and the Two Axis Valuator [5]. They found it more efficient than Screenspace, and comparable to StickyTools. Their technique can be performed with two fingers of the same hand, and thus it is applicable to mobile devices. However, the technique does not allow the simultaneous control of the 3 rotation axis: it requires to switch between modes, similarly to the Two Axis Valuator + Z proposed by Zhao *et al.* [14].

Scheurich *et al.* proposed an adaptation of the Two Axis Valuator + Z for multi-touch interaction [10]. In this new technique, the control of translation is added so as to create a full 6 DOF manipulation technique. Here again, two separate modes control the various rotations and do not allow integral control.

Multi-touch 3D interactions, including rotations, have also become more common in commercial products on smart phones and tablet computers. For example, the two software 123 Sculpt from Autodesk™and Cortona 3D™offer a virtual trackball interaction with one finger, and the control of the Z rotation with two fingers. The two modes being exclusive, it is not possible to control the 3 axis of rotation at the same time.

2.3 Integral control

Integral control, i.e. the simultaneous control of multiple DOF, has long been seen as a way to improve the efficiency of interaction techniques. However, users' cognitive capabilities in terms of motor control have limits. These limits are observable when asking users to control too many DOF, or to use an interaction technique that presents an unsuitable mapping between the input and the output.

Hinckley *et al.* compared the performance of the Virtual Sphere and the Arcball with that of a free-moving magnetic device allowing the integral control of 3 DOF rotations [4]. They found that users were more efficient with the integral control, with no loss of precision. Later, Masliah and Milgram studied the allocation of control on a 6 DOF docking task using 6 DOF input devices [8]. They designed the m-metric to measure the integration of control. The m-metric revealed that 6 DOF was too much for the participants, as they separated the control between two groups of DOF: the 3 translations and the 3 rotations. Still, this study indicates that users are able to control the 3 rotations in an integral way. These two studies used input devices that have a direct mapping between the physical rotation applied to the device and the resulting rotation of the virtual object. This direct mapping is not possible on the flat multi-touch surface of most mobile devices.

More recently, Martinet *et al.* studied integration of control using a large multi-touch display [6]. They confirmed the study of Masliah and Milgram as they found that the integration of the six degrees of freedom was less efficient than separating rotation and translation. They proposed a new 6 DOF technique, named *DS3*, that is based on Screenspace for the control of rotation. DS3 was found more efficient than Screenspace and StickyTools but, using Screenspace for rotations, it suffers from the same flaws: unpredictable behavior and some unreachable orientations.

In summary, integral control of 3 DOF rotation has been shown to be more efficient than separate control, but only with a direct mapping of input to output. As a direct mapping is not possible on a flat multi-touch surface, the ability of users to perform integral control efficiently remains to be shown.

3. THE SURJECTION CRITERION

As seen in the literature review, some interaction techniques separate the control of the 3 DOF of 3D rotations either by using different devices (e.g. the mouse wheel controls the Z-rotation [14]), or using spatial modes to define which DOF are controlled (e.g. the initial contact position of the Arcball [12]). The consequence is that some orientations of the controlled object cannot be reached by a single *gesture* on the input device.

For example, with the Arcball technique, the *control axis* is defined as the axis passing through the center of the Arcball and the initial contact point. Once the mouse, or the finger, is controlling the Arcball and the control axis is defined, rotations of the object around the control axis are unreachable. When simple rotations must be applied to the controlled object, users can usually take care to make the initial contact so that the target orientation is reachable. For more complex rotations however, choosing the correct initial contact may not be possible. The side of the object that the user wants to see may be facing the back of the screen, for example, and thus its orientation (upright or upside down) can not be perceived. As a consequence, the desired rotation is unknown until the interaction has started, the desired face has been brought to sight, and the user can see which rotation in the screen plane is needed. Hence the Arcball interaction is frequently decomposed into a sequence of atomic gestures which are interrupted when a new control axis must be defined and the mouse, or the finger, have to unclutch and then clutch again.

We generalize this problem by stating that the Arcball interaction does not satisfy the *surjection* criterion, i.e. it is not surjective. We define that a surjective interaction technique gives the possibility to reach any final state of the controlled object from any current state of an ongoing interaction of this technique. The name of this criterion is inspired by the the mathematical concept: a surjection is a function f defined on a set A and taking values in a set B such that for any $b \in B$ there exists an $a \in A$ for which $f(a) = b$. In the case of interaction techniques, B corresponds to the set of all possible *state* that the controlled object can take. The set A corresponds to the set of all possible sequence of events of a *single gesture* of the interaction. Such sequences typically only contain "motion" events, and no "mouse-button-up", "mouse-button-down", "finger-touch", or "finger-untouch" events. f_{b_c} corresponds to the *transfer function* of the interaction technique, which maps an input a to a final state of the controlled object b_f starting from a current state b_c.

An interaction technique satisfies the surjection criterion if for any current state of the controlled object b_c, any final state b_f, there is a sequence of motion events a that moves the object from b_c to b_f. In mathematical form:

$$\forall b_c, \forall b_f, \exists a : f_{b_c}(a) = b_f \tag{1}$$

A surjective interaction technique allows users to adjust their input in a *continuous* manner while working to reach their objective. As such, surjection can be seen as a key criteria to realize the 3rd principle of the "Direct Manipulation" paradigm. This principle states that operations must be "rapid, incremental, reversible" [11]. When comparing *non*-surjective interactions to surjective interactions, the former are less reversible in the sense that an initial error can not be corrected by further adjustments. They are less incremental at the gesture level, as some rotations may require the combination of several gestures, i.e. the goal is unreachable by additional correction to a single gesture. Finally, they are less rapid in the sense that stopping a gesture and starting a new one is generally more costly than a continual adjustment of a single gesture.

As direct manipulation remains a fundamental guideline to create easy to learn and efficient interactions, we put a strong emphasis on designing interaction techniques which are surjective.

4. 3D ROTATIONS WITH TWO FINGERS

We developed two new techniques for 3D rotations in the context of single-handed multi-touch interactions. The first one extends Scheurich *et al.*'s Two Axis Valuator + Z, we call it 'TAV+". The second one extends Shoemake's Arcball, we call it "Arcball+". These adaptations use two fingers of a single hand. Both allow the integral control of the 3 DOF and are surjective.

4.1 TAV+

4.1.1 Original Two Axis Valuator +Z

Scheurich's Two Axis Valuator + Z (*TAVZ*) uses two rotation modes triggered by the number of fingers in contact with the multi-touch device [10]. One finger controls the rotation around an axis defined in the screen plane and perpendicular to the movement (see Figure 1a). The amount of rotation is tightly linked to the movement distance. In the basic version of TAVZ this amount is equal to 180 degrees when the finger travels from the left side of the screen to the right side, but it is customizable.

Two fingers control the *Z rotation*: the first finger defines the *anchor point* and remains fixed, the second finger gives the amount of rotation (see Figure 1b). The rotation axis is perpendicular to the screen plane and passes through the anchor point.

The two modes introduce a discrepancy in the *pivot point* of the rotation: the one-finger mode applies rotation to a pivot point previously defined by the user while the two-fingers uses a temporary *pivot point*. When the two pivot points are not on the same location, the two-fingers mode results in a combination of rotation and translation of the object. In some situation, this moves the object out of the screen viewport, and it becomes difficult to bring it back. Being able to define the pivot point adds complexity to the interaction, and the benefit is unclear in the case of simple observation tasks for novice users.

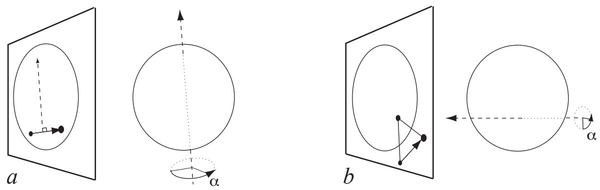


Figure 1: TAVZ. *a*: with 1 finger, the rotation axis is defined in the screen plane, and perpendicular to the finger movement. *b*: with 2 fingers, the rotation axis is perpendicular to the screen, and passes through the anchor point defined by the first finger.

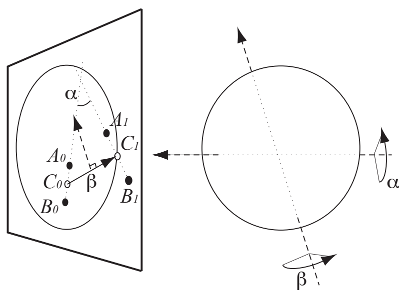


Figure 2: TAV+: X and Y rotations are controlled by the position of the middle of the two finger positions (C_i). The Z rotation is controled by the rotation of the vector made by the two fingers.

4.1.2 *Improving TAVZ*

We designed an extension of TAVZ that satisfies the following constraints: the interaction should not have modes, it should always rotate the object at a fixed pivot point (the object's center of gravity), it should be surjective, and it should allow integral control of the 3 DOF. We called this interaction TAV+. It is only operated with two fingers. Similarly to TAVZ, the X and Y rotations are controlled by the position of one control point: the center of the two fingers contact points. The Z rotation is linked to the rotation of the vector made by the two fingers. As illustrated in Figure 2, let A_0, B_0 the starting positions of each finger at the beginning of the movement and A_1, B_1 their ending positions. Let C_0, C_1 the respective middles of $[A_0, B_0]$ and $[A_1, B_1]$. The new orientation of the object is defined by the combination of two rotations:

1. one rotation around the axis perpendicular to the screen, passing through the object's center. The amount of rotation is defined by the angle between $[A_0, B_0]$ and $[A_1, B_1]$.

2. the other rotation around the axis perpendicular to $[C_0, C_1]$, parallel to the screen, and passing through the object's center. The amount of rotation is defined as in Scheurich *et al.*'s implementation: 180 degrees when dragging from the left to the right side of the screen.

In the Section 5, we provide a comparative analysis of the criteria that are satisfied by TAV+.

4.2 Arcball+

4.2.1 *Original Arcball*

Shoemake's Arcball is defined with the use of an invisible sphere which surrounds the controlled object [12]. The orientation of the object is linked to the orientation of the sphere, so users only manipulate the sphere. Being designed for a 2 DOF input device (e.g the mouse), rotation modes are triggered by spatial activation: interaction *on* or *out* of the sphere.

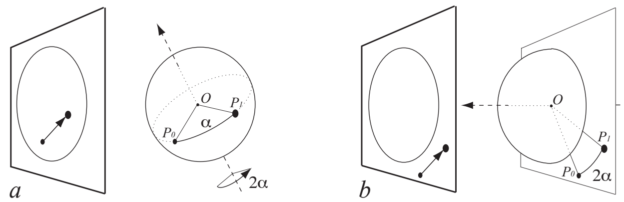


Figure 3: Arcball: *a*: interaction with 1 finger onto the sphere allows the integral control of the 3 rotation axis. *b*: outside of the sphere, the finger controls a rotation around a normal to the screen.

The mouse or finger screen location is projected onto the sphere. Let P_0 and P_1 be the projections of the starting and the ending positions of the 2D input device. O is the position of the center of the sphere. The rotation axis is then the cross product of $\overrightarrow{O,P_0}$ and $\overrightarrow{O,P_1}$. This axis is perpendicular to the plane defined by $\overrightarrow{O,P_0}$ and $\overrightarrow{O,P_1}$, and passes through O. When the projection stays on the sphere, the rotation axis is a combination of the 3 axis X, Y and Z, as shown on Figure 3a. As such, Arcball allows an integral control of the 3 DOF. A constraint from the Shoemake's algorithm makes the amount of rotation be twice the angle between $\overrightarrow{O,P_0}$ and $\overrightarrow{O,P_1}$. If the projection does not intersect the sphere, it is projected on the plane that is parallel to the screen and passes through the center of the sphere, as illustrated on Figure 3b. The rotation axis is then perpendicular to the screen. Because of the two modes (on and out of the sphere), the rotation is partially controlled in an integral way: it is integral only when the projection remains on the sphere. Another limitation of Arcball is that it does not satisfy the surjection criteria, as explained in section 3.

4.2.2 *Improving Arcball*

We introduce a surjective and entirely integral version of the Arcball which allows a customizable amount of rotation around the "Control Axis". We call this interaction "Arcball+". It is always operated with two fingers. Let C be the middle of the two contact points. pC is the projection of C onto the sphere. pC is used to manipulate the sphere in the same way as in the original Arcball. Similarly to TAV+, the rotation of the contact points provides a new DOF which is used for in the control of the rotation around the Control Axis (O, pC), as illustrated on Figure 4. This is the main difference with the original Arcball; it makes Arcball+ a surjective interaction technique. In Arcball+, the amount of rotation is tightly linked to the movements of the projected points, but it is customizable unlike the Arcball.

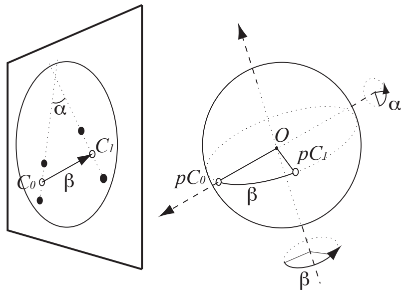


Figure 4: Arcball+: The projection of the middle of contact points pC controls the sphere as in the original Arcball. The rotation of the vector defined by the two contact points controls the rotation around the axis defined by O and pC_0.

5. CRITERIA ANALYSIS

Table 1 summarize how each of the previously described techniques satisfy the ergonomics, integrality and surjection criteria.

Ergonomics criteria from Bade *et al.* [1]:

- ***B1***: TAV+ is the only technique that respects the first criteria, as it is the only that provides the same outcome regardless of the initial location of an interaction.

- ***B2***: All the techniques match the direction of pointing and the direction of rotation.

- ***B3***: Neither TAVZ nor TAV+ respect the transitivity principle. This is linked to the basic principle of TAV, in which the rotations are always applied to a global frame that does not move with the object. Rotating an object by $+90°x$, $+90°y$, $-90°x$, and then $-90°y$ does not bring it back as it started, but off by 120°.

- ***B4***: The Arcball is the only technique that forces a fixed C/D ration. This constraint is freed in Arcball+ as it allows a customizable control-to-display ratio.

Integrality (*I*):

Arcball partially respects the integrality criteria: simultaneous control is possible on the 3 DOF, but only when the device location projects onto the sphere. This criterion is preserved in Arcball+, but also extended to the whole interaction area: Arcball+ is fully integral.

TAVZ has modes, which prevents the simultaneous control of the 3 DOF. This is corrected in TAV+.

Surjection (*S*):

Arcball is not surjective because it does not allow to reach any new object orientations from any contact point: the only reachable rotations are the ones defined by the axes which lie in a plane normal to the control axis (see section 3).

Arcball+ adds a new DOF of input to Arcball. The additional DOF allows rotations around the control axis. Any desired rotation axis can thus be decomposed into one axis reachable by Arcball and this additional control axis. At any point of the interaction, it is thus possible to find a trajectory to reach any new object orientation. Arcball+ is thus surjective.

Because of the rotation modes, TAVZ is not surjective. Users have to switch between modes, hence to decompose rotations in order to reach their goal.

TAV+ allows to manipulate the 3 rotation axes at the same time, but also to manipulate each of them independently. It is thus possible to reach any new object orientation from the current one: TAV+ is surjective.

Table 1: Availability of ergonomics, integration and surjection criteria for both basic and new techniques

	Arcball	Arcball+	TAVZ	TAV+
B1	-	-	-	+
B2	+	+	+	+
B3	+	+	-	-
B4	-	+	+	+
I	-/+	+	-	+
S	-	+	-	+

6. EXPERIMENTAL EVALUATION

We conducted a user experiment in order to evaluate the effect of the interaction technique and of the surjection criterion on user efficiency and integration of control.

6.1 Task and Apparatus

The experiment was run using an ad-hoc application running on a laptop (Intel Core i7, 2.6 GHz, NVidia GT 650M). We used a multitouch display (Wacom Cintiq 24HD touch) at its native screen resolution: 1920x1200 pixels. As a vertical display configuration generates a lot of arm fatigue, the display was setup with 30° of angle from horizontal. Participants interacted with the display with one or two fingers. They also controlled the different steps of the experimentation by pressing the space bar and the return key on an external keyboard.

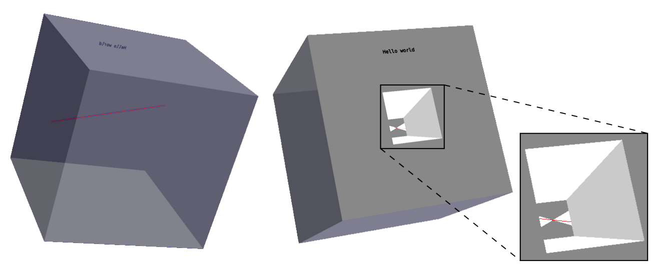


Figure 5: The Cube. Left: initial random orientation, the target face is on the back, the text is seen in reverse through transparency. Middle: the hole is facing the screen within the 20° tolerance for the y axis (top and bottom side of the hole visible) and the z-axis (red line within white area), but not for the x-axis (left side of the hole not visible). Right: close-up of the hole.

Participants had to perform a docking task. A cube with a random initial orientation was displayed, with a size of about 75% of the height of the display. The cube had a square hole on one of its faces. Participants had to rotate the cube so that the hole appeared in front of the display. As illustrated on Figure 5, we used the sides of the hole and a spirit level to provide an intuitive feedback of the tolerance along the three rotation axis. The cube was docked within the requested tolerance if all sides of the hole were visible, and the red line of the spirit level was within the white area. The hole shape and white area were adapted for the desired tolerance levels. Participants validated a docking by pressing the space bar on the keyboard. They pressed the spacebar a second time to initiate a new docking.

We used two tolerance levels in the experiment: 2° and 20°. The 2° tolerance level was used to model a *high precision* task such as doing precise alignments of a virtual object when modeling a scene. The main motivation of this work, however, was to study the general exploration of an object, hence we included a *low precision* task (20° of requested tolerance) in the experiment.

6.2 Participants

We recruited 16 unpaid participants, aged in the range of 23 to 57, 7 women and 9 man; 2 were left-handed, 13 right-handed, and one was ambidextrous; 7 were university students and 9 were not. 14 participants used multitouch devices such as smart phones on a daily basis. 2 participants had once manipulated a 3D scene and 14 had never manipulated any 3D scene before.

6.3 Techniques

The two new techniques, TAV+ and Arcball+, are compared to Arcball and TAVZ. However, in order to better fit with an observation task, and to better respect the ergonomics criteria of Bade *et al.*, Arcball and TAVZ have been slightly adapted:

1. We adapted TAVZ by applying all the rotations to the object's center. This prevents any combination of rotation and translation implied by the two-fingers mode (see section 4.1.1). We call this adaptation TAV^2.

2. We added a two-fingers mode to the Arcball. With only one finger this adaptation is equal to the basic Arcball, but when a second contact point appears, the user can control the rotation around the control axis, made by the center of the sphere and the projected first point. Similarly to Arcball+, we called the adaptation Arcball^2.

Figure 6 sums up all the techniques that we describe in this paper, and the links between each of them.

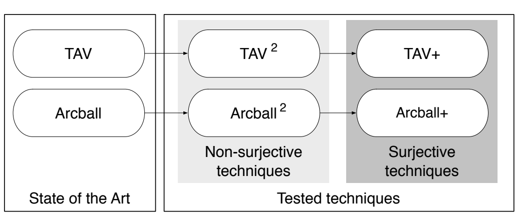


Figure 6: Evolutions from Arcball and TAVZ. In order to match to observation tasks, both Arcball and TAVZ have been slightly adapted before being compared with Arcball+ and TAV+.

6.4 Procedure

The study was conducted in a silent room. Participants were sitting in front of the multitouch screen. The investigator stayed during the whole experiment. At the beginning of a session, the investigator explained the procedure to the participant and performed a demonstration of the interactions.

Each participant performed dockings using each of the 4 interaction techniques: TAV^2, Arcball^2, TAV+ and Arcball+. Every technique was tested for the high and low requested precisions, with 20 repetitions of each. Overall, we measured: 16 (participants) x 4 (interaction techniques) x 2 (requested precisions) x 20 (repetitions)= 2560 dockings.

We measured both completion time and achieved precision. Completion time was registered between the first and the last touch before validation. Achieved precision was measured as the minimal rotation required to rotate the cube from the participant's final position to the perfectly aligned position.

Each new combination of technique and accuracy started with a training phase allowing participants to perform an unlimited number of dockings. When they felt confident, they pressed the "return" key on the keyboard to start the real test. The order of presentation of tolerance was balanced between 2 groups of 8 participants. Within each groups, we balanced the presentation order of the interaction techniques.

In a post-experiment interview, participants sorted the interaction techniques according to their preferences on 3 criteria: general preference, speed, and precision.

6.5 Results

Table 2: Results of the experiment

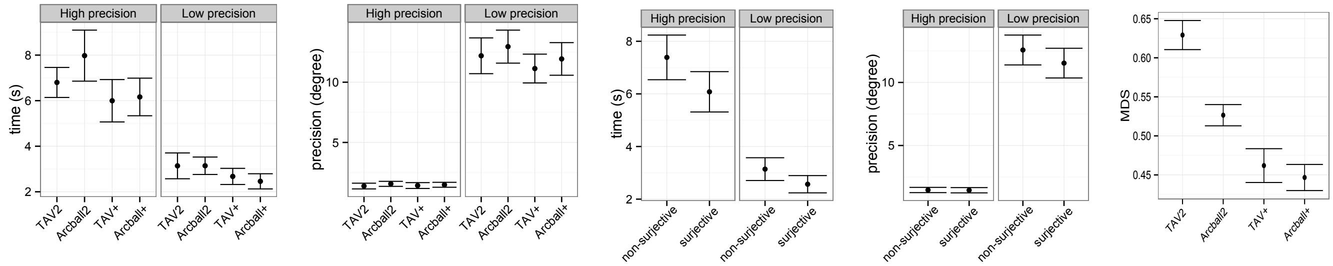
	Accomplishment time (s.)		Achieved precision (°)	
Requested precision	high (2°)	low (20°)	high (2°)	low (20°)
Interaction technique				
TAV^2	6,80	3,14	1,38	12,19
Arcball^2	7,98	3,14	1,56	12,95
TAV+	6,00	2,67	1,42	11,13
Arcball+	6,16	2,46	1,48	11,93
Surjection				
non-surjective	7,39	3,14	1,47	12,57
surjective	6,08	2,57	1,45	11,53

Figure 7 shows the results of the experiment, the corresponding values are presented in Table 2. We report the results on the effect of the interaction technique for each individual interaction technique, but we also report on the effect of the surjection criterion: TAV^2 and Arcball^2 constitute the "non-surjective" group, TAV+ and Arcball+ the "surjective" group. The measures reported for each group are simply the average of the measures of its two techniques.

We also computed a measure of the amount of integral control that could be observed in participants' interactions. We used the *Magnitude of DOF Separation* (MDS) measure as defined by Veit *et al.* [13]. An MDS value of 0 indicates that all DOF are manipulated over a time window, while a value of 1 indicates that a single DOF was manipulated during a time window. Unfortunately, the late discovery of an error in the code that logged the trajectories prevented us to compute the MDS measures for all the participants: only the data from the last 8 participants was usable. The MDS values are shown on Figure 7e.

6.5.1 *Effect of the interaction technique*

A one way repeated measure ANOVA revealed that the interaction technique had an effect on accomplishment time,



a) b) c) d) e)

Figure 7: Results of the experiment. a) and b): average over all participants of the task accomplishment time and precision, respectively, as a function of the interaction technique, with 95% confidence intervals. c) and d): the same measures, averaged by surjection. High precision is 2° of tolerance, low precision is 20°. e) average over 9 participants of the MDS measure of integral control (0 is entirely integral).

both when the requested precision was high ($F(3,45) = 9.84, p < 10^{-4}$) and low ($F(3,45) = 8.47, p < 10^{-3}$). Pair-wise comparisons using paired sampled t-tests with Bonferroni correction revealed a significant time difference at high precision for Arcball^2 versus TAV^2 ($p < 0.05$) and Arcball^2 versus TAV+ or Arcball+ ($p < 0.01$), but no other significant differences. The Arcball^2 technique was the least efficient. At low precision, the only significant time differences were between TAV^2 versus Arcball+ ($p < 0.05$)), Arcball^2 versus TAV+ ($p < 0.05$) and Arcball^2 versus Arcball+ ($p < 0.01$).

The ANOVA did not reveal any effect of the interaction technique on achieved precision, neither at high nor at low precision. This is illustrated by the very similar distributions on Figure 7b.

The ANOVA did not find a significant *general* effet of the interaction on the MDS measure of integral control, despite a low probability of the NULL hypothesis ($F(3,24) = 2.40, p = 0.0929$). Had we been able to capture the interaction trajectories of all our participants, a significant effect would have been likely. In any cases, a pair-wise comparisons using paired sampled t-tests with Bonferroni correction confirmed what is visible on Figure 7e: there was a strongly significant difference between each pair of interaction techniques ($p < 0.01$), except the two surjective techniques. With the 2 surjective interactions, participants executed 14% more integral control compared to Arcball^2 and 28% more integral control compared to TAV^2.

6.5.2 *Effect of surjection*

A two-tailed paired sample t-test revealed at both the high and low precisions that the integration capability of an interaction technique had an effect on task accomplishment time ($t(15) > 4.4, p < 10^{-3}$). At both precisions, the surjective interaction techniques were about 18% more efficient than the non-surjective techniques. A two-tailed paired sample t-test did not show any significant effect of the integration capability of an interaction on the achieved precision, neither at high nor at low precision.

6.5.3 *Subjective evaluation*

Participant's preferences where coded by giving a score in the set $\{1, 2, 3, 4\}$ to each interaction, 4 being the preferred interaction. Figure 8 shows the resulting subjective ratings averaged over the 16 participants. Friedman rank sum test performed on the data revealed that there was a significant effect of the interaction technique on participant's preferences, for the 3 criteria (Friedman chi-squared $> 15.9, df = 3, p < 0.01$). We performed pairwise comparisons using Wilcoxon rank sum test with the Bonferroni adjustment method for each criteria. For the general preferences, the only significant difference were between Arcball+ on one side, and TAV^2 and Arcball^2 on the other side ($p < 0.01$). For the precision preferences, significant differences where found between each surjective techniques (TAV+ and Arcball+) on one side, and each non-surjective techniques on the other side (TAV^2 with $p < 0.05$ and Arcball^2 with $p < 0.01$). The same pattern was observed for the speed preferences, with significant differences found between each surjective techniques on one side, and each non-surjective techniques on the other side ($p < 0.01$). In all cases where the differences were significant, the surjective techniques were preferred compared to the non-surjective techniques.

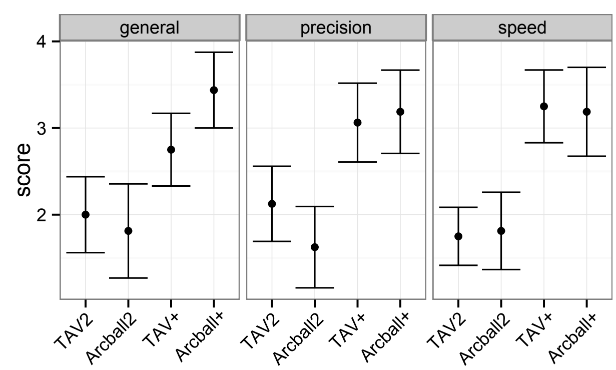


Figure 8: Subjective ratings of each interaction technique for the 3 criteria. Interactions with higher score are preferred.

6.6 Discussion

The main result of this experiment is that participants were, on average, 18% faster using the Surjective and integral interaction techniques. This improvement in task accomplishment time did not seem to degrade participants' precision, as we did not find any significant difference in precision. It should be stressed that all of our participants did not have prior experience with 3D interaction, and half of them did not have any training in computer science. As a result, this experiment provides a strong evidence that surjection and integrated control of the 3 axes of rotation are indeed usable on flat multi-touch surfaces, and notably improve performance. Even though the experiment was executed on a large interactive surface, this result should transfer directly to mobile devices as all of the tested interaction techniques were executed with 2 fingers of the same hand.

A second result of the experiment is about the usage of integrality (Figure 7e). Providing integral interaction techniques does not mean that users will, and will be able to, exploit it. TAV^2 is not an integral technique, so it is logical that its MDS result is far from 0 (0 means that all DOF are manipulated).
Arcball and Arcball^2 have been considered as partially integral techniques. This can explain why the results of the MDS measure is close, but slightly up, to 0.5.
The best MDS results are with TAV+ and Arcball+ (not significantly different). With these techniques users can exploit the integrated control offered. These results confirm our expectations. However, as the techniques are surjective, and allow to continuously adjust the trajectory for reaching the final orientation, users sometimes manipulate 2 or only 1 DOF. Moreover, the docking task could be decomposed into two sequential phases: traveling and adjustment, and we suppose that the fine adjustment needs less integrality than the first phase. These two aspects could explain why the value is not closer to 0. They will be investigated in future work.

Finally, participants expressed their preference towards the surjective interaction techniques. Arcball+ was particularly appreciated, as it ranked consistently higher than the other techniques. TAV+ ranked almost as good as Arcball+. But, in the "general" preference, its ranking was not significantly higher than the ranking of surjective techniques. Beyond these results, surjection and integration have been distinctly perceived and appreciated by the participants. They indeed found that it was easier to perform the task "without having to switch between modes". Participants who preferred Arcball+ towards TAV+ expressed that the movement of the object during rotation was more "natural".

7. CONCLUSIONS

In this paper we introduced a new criterion for interaction techniques: surjection. We used surjection, as well as integrality and other ergonomics criteria, to design 2 new rotation techniques: Arcball+ and TAV+. We ran a user experiment in which both quantitative and qualitative results demonstrated the benefits of these techniques. Participants were able to perform a more integral control of all 3 axes of rotation, and thus improved their performances compared to classical techniques. As the participants had no prior experience on 3D interaction, these results indicate that these techniques are suitable for applications aimed at the novice users.

A natural extension of this work will be to implement the techniques on tablets and smart phones, in order to conduct a longitudinal study. This study will allow to test the techniques on observation tasks in "mobile" conditions: while walking or riding a bus for instance. In parallel, we will study user gestures during observation task in more details, in order to get a deeper understanding of the benefits of integrality and surjection.

8. AKNOWLEDGMENTS

This work was partially funded by the French government in the project 3DCI (FUI-AAP14) and the LabEx PERSYVAL-Lab (ANR-11-LABX-0025-01).

9. REFERENCES

[1] Bade, R., Ritter, F., and Preim, B. Usability comparison of mouse-based interaction techniques for predictable 3d rotation. *Smart graphics* (2005).

[2] Chen, M., Mountford, S. J., and Sellen, A. A study in interactive 3-d rotation using 2-d control devices. *In Proc. of SIGGRAPH CG 22*, 4 (1988), pp 121–129.

[3] Hancock, M., ten Cate, T., and Carpendale, S. Sticky tools: full 6dof force-based interaction for multi-touch tables. In *Proc. of ITS*, ACM (2009), pp 133–140.

[4] Hinckley, K., Tullio, J., Pausch, R., Proffitt, D., and Kassell, N. Usability analysis of 3D rotation techniques. *In Proc. of UIST* (1997), pp 1–10.

[5] Liu, J., Au, O. K.-C., Fu, H., and Tai, C.-L. Two-Finger Gestures for 6DOF Manipulation of 3D Objects. *Computer Graphics Forum 31*, 7 (Sept. 2012), pp 2047–2055.

[6] Martinet, A., Casiez, G., and Grisoni, L. The effect of dof separation in 3d manipulation tasks with multi-touch displays. In *Proc. of VRST*, IEEE (2010), pp 111–118.

[7] Martinet, A., Casiez, G., and Grisoni, L. Integrality and separability of multitouch interaction techniques in 3d manipulation tasks. *IEEE TVCG 18* (2012), pp 369–380.

[8] Masliah, M. R., and Milgram, P. Measuring the allocation of control in a 6 degree-of-freedom docking experiment. In *Proc. of CHI*, ACM (2000), pp 25–32.

[9] Reisman, J. L., Davidson, P. L., and Han, J. Y. A screen-space formulation for 2D and 3D direct manipulation. In *Proc. of UIST*, ACM (Oct. 2009), pp 69.

[10] Scheurich, D., and Stuerzlinger, W. A one-handed multi-touch mating method for 3d rotations. *In Proc. of CHI, Ext. Abstract* (2013), pp 1623.

[11] Shneiderman, B. Direct manipulation: A step beyond programming languages. *Computer 16*, 8 (aug. 1983), pp 57 –69.

[12] Shoemake, K. Arcball: a user interface for specifying three-dimensional orientation using a mouse. In *Proc. of GI*, Morgan Kaufmann Publishers Inc. (1992), pp 151–156.

[13] Veit, M., Capobianco, A., and Bechmann, D. Influence of degrees of freedom's manipulation on performances during orientation tasks in virtual reality environments. In *Proc. of VRST*, ACM (2009), pp 51–58.

[14] Zhao, Y. J., Shuralyov, D., and Stuerzlinger, W. Comparison of multiple 3d rotation methods. In *Proc. of VECIMS* (2011), pp 1–5.